COMP 202 - Spring 2021 - Homework #3

Due date: Wednesday, May 5, 2021 at 23:59

HW Description:

Implement a Binary Search Tree using pointer based implementation.
1. Write a function to **insert** a new item in the tree. If the item is already present, do nothing.
2. Write a function to check whether the tree **is balanced** or not. A binary tree is balanced if the height of the left and right subtree of any node differ by not more than 1.
3. Write a function to **remove** an item from the tree. If the item is not present, do nothing.
4. Write a function to **compare** two binary trees. It should return true if the two trees have the same items at the corresponding nodes, false otherwise.

Submission Materials:
In all homeworks, you are required to submit two things:
- A pdf file with your pseudocode and asymptotic complexity analysis (time and space). If it is handwritten, make sure it is legible.
- A java implementation that compiles and runs without additional setup (submitted through github classroom **and** via Blackboard).

In your pdf file, you must copy-paste the following statements exactly, and sign it by hand. You can use your phone camera for that. Without this signed statement, your submission will not receive any grade.

I have completed this assignment individually, without support from anyone else. I hereby accept that only the below listed sources are approved to be used during this assignment:

(i)   Course textbook,

(ii) All material that is made available to me by the professor (e.g., via Blackboard for this course, course website, email from professor / TA),

(iii) Notes taken by me during lectures.

I have not used, accessed or taken any unpermitted information from any other source. Hence, all effort belongs to me.

Github Submission:
Usage of github classroom and submission details:
1) Accept the invitation using [this link](https://classroom.github.com/a/s1_rjCdm) (https://classroom.github.com/a/s1_rjCdm)
    a) If you have no github account, create one.
2) Choose your student ID from the list, it will be linked to your github account from now on.
3) A personal repository with the starter code will be created. You can directly edit the code on github's page, clone it to your local storage and edit it there, or use the online IDE Repl.it.
4) Make sure your changes are committed and pushed to the repository.
5) Check if you have successfully passed the tests in the "Actions" tab of your repository. Note that these automatic tests help you understand whether or not your solution works up to some level, but passing them does not guarantee that you will receive a full grade.
6) To make sure your code is received and avoid any potential problems on github's side, submit a copy of java files together with the above-mentioned pdf on Blackboard as well. Submit all files as a single .zip file, named as: "ID_NAME_SURNAME.zip"
7) This is an individual project.
8) If you make the github project public instead of private, it will be counted as cheating.
9) No late submissions are accepted.